A Survey of Large Language Model Applications: Taxonomy, Benchmarks, and Emerging Directions

Anonymous

January 31, 2026

Abstract

Large Language Models (LLMs) have transitioned from pure language modeling to a general-purpose reasoning engine that can be applied across a wide spectrum of domains—from code synthesis to scientific discovery and multimodal generation. This paper provides a systematic survey of LLM applications published on arXiv in the last two years (2022–2024). We propose a taxonomy that groups applications into five high-level categories, discuss the evaluation benchmarks and metrics that have emerged, and highlight the principal challenges such as hallucination, privacy, and computational cost. Finally, we identify promising research directions, including retrieval-augmented generation, chain-of-thought prompting, hybrid symbolic–neural systems, and continual adaptation. The survey aims to serve both newcomers seeking an overview and experts looking for concrete open problems.

1 Introduction

The past few years have witnessed a rapid escalation in the scale and capability of LLMs, exemplified by models such as GPT-4 [OpenAI, 2023], LLaMA [Touvron et al., 2023], and Mistral [Jiang et al., 2023]. While the original objective of these models was next-token prediction, researchers have demonstrated that, with appropriate prompting or fine-tuning, LLMs can perform tasks that were traditionally reserved for specialized systems. Consequently, a plethora of application papers have appeared on arXiv, covering domains as diverse as software engineering, healthcare, finance, and multimodal reasoning.

In this survey we answer three questions:

i) What are the major categories of LLM applications and how do they differ in terms of input/output modalities and required reasoning depth?

ii) Which benchmarks and metrics are used to evaluate these applications, and what are their limitations?

iii) What are the most promising research directions that can address current shortcomings?

Our contributions are:

- A concise taxonomy of LLM applications (Section 3).
- A curated list of benchmark suites and a formalization of common evaluation metrics (Section 4).
- An analysis of open challenges and a forward-looking research agenda (Section 6).

2 Background

LLMs are trained by minimizing the cross-entropy loss over a large corpus of text. Given a sequence of tokens $\mathbf{x} = (x_1, \ldots, x_T)$, the objective is

$$\mathcal{L}(\theta) = -\sum_{t=1}^{T} \log p_\theta(x_t \mid x_{<t}), \tag{1}$$

where p_θ is the model parameterized by θ. The per-token perplexity $\text{PPL} = \exp\left(\frac{1}{T}\mathcal{L}(\theta)\right)$ is a standard proxy for model quality.

Recent advances have focused on scaling (e.g., 175B parameters in GPT-3 [Brown et al., 2020]), instruction tuning [**?**], and reinforcement learning from human feedback (RLHF) [**?**]. These techniques have dramatically improved zero-shot and few-shot performance, enabling LLMs to be repurposed for downstream tasks via prompting.

3 Taxonomy of LLM Applications

We organize LLM applications into five categories, each characterized by its primary modality, typical downstream task, and representative works.

3.1 Natural Language Processing (NLP)

Traditional language tasks such as summarization, translation, and question answering now rely on prompt engineering and instruction-tuned models. Notable examples include chain-of-thought prompting for arithmetic reasoning [Wei et al., 2022] and retrieval-augmented generation (RAG) for knowledge-intensive tasks [Lewis et al., 2020].

3.2 Code Generation and Software Engineering

LLMs such as Codex [Chen et al., 2021] and CodeLlama [Rozière et al., 2023] can synthesize code from natural language specifications, perform automated debugging, and generate unit tests. Recent work on hybrid attention architectures (HALO) [Chen et al., 2024] further improves long-context reasoning for large code bases.

3.3 Scientific Discovery and Quantitative Domains

Models have been applied to hypothesis generation in biomedicine [Zhang et al., 2023], drug repurposing [**?**], and quantitative finance [**?**]. The RedSage project [Suryanto et al., 2024] demonstrates domain-specific continual pre-training for cybersecurity.

3.4 Multimodal Generation

Unified models that generate both images and text, such as UEval [Li et al., 2024] and Flamingo [**?**], require joint vision-language representations. Evaluation now relies on rubric-based scoring rather than simple LLM-as-judge methods.

3.5 Decision Support and Knowledge-Intensive Tasks

LLMs are increasingly used as assistants in healthcare, finance, and education, where they must retrieve up-to-date information and provide explanations. Retrieval-augmented pipelines (e.g., RAG) and tool-use agents (e.g., AutoGPT) are central to this category.

4 Benchmarks and Evaluation Metrics

Evaluating LLM applications is challenging because the output space is often open-ended. Below we summarize the most widely used benchmarks and formalize the associated metrics.

4.1 Standard NLP Benchmarks

Datasets such as GLUE [**?**], SuperGLUE [**?**], and MMLU [**?**] provide multiple-choice or short-answer tasks. Accuracy $\text{Acc} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}[\hat{y}_i = y_i]$ remains the primary metric.

4.2 Code Benchmarks

HumanEval [Chen et al., 2021] and MBPP [**?**] assess functional correctness by executing generated code against unit tests. The pass@k metric [Chen et al., 2021] is defined as

$$\text{pass@k} = \frac{1}{N}\sum_{i=1}^{N}\left(1 - \frac{\binom{n_i - c_i}{k}}{\binom{n_i}{k}}\right), \tag{2}$$

where n_i is the number of samples for problem i and c_i the number of correct samples.

4.3 Scientific and Domain-Specific Benchmarks

RedSage-Bench [Suryanto et al., 2024] (30K MCQ + 240 open-ended) and MATH [**?**] evaluate reasoning depth. For multimodal tasks, UEval [Li et al., 2024] provides rubric-based scores across 10,417 criteria.

4.4 Emerging Metrics

- **Faithfulness**: KL divergence between model-generated distribution and a reference distribution, $\text{KL}(p\|q)$.
- **Hallucination Rate**: proportion of generated statements not supported by external knowledge bases.
- **Tool-Use Success**: binary success indicator for agentic pipelines, aggregated as $\text{Success} = \frac{1}{M}\sum_{j=1}^{M} \mathbf{1}[\text{task}_j \text{ completed}]$.

5 Challenges and Limitations

1) **Hallucination and Reliability**: LLMs may generate plausible-looking but false statements, especially in high-stakes domains.

2) **Data Privacy and Security**: Deploying LLMs locally (e.g., RedSage) mitigates leakage but raises concerns about proprietary data contamination.

3) **Computational Cost**: Even inference for 8B-parameter models can be prohibitive on commodity hardware.

4) **Evaluation Bottlenecks**: Open-ended tasks lack universally accepted metrics; rubric generation is labor-intensive.

5) **Alignment and Bias**: RLHF improves helpfulness but does not fully eliminate harmful biases.

6 Future Research Directions

Based on the surveyed literature, we propose four high-impact research avenues.

6.1 Retrieval-Augmented Generation (RAG) with Dynamic Knowledge Bases

Current RAG pipelines use static document stores. A formal model can be expressed as

$$\hat{y} = \mathrm{Dec}\big(\mathrm{Enc}(x),\ \mathrm{Retr}(x, \mathcal{K}_t)\big), \tag{3}$$

where $\mathcal{K}_t$ evolves over time. Developing efficient update mechanisms and consistency guarantees is an open problem.

6.2 Chain-of-Thought and Self-Consistency at Scale

Extending chain-of-thought prompting to multi-turn dialogues and tool use can improve reasoning depth. Research is needed on sampling strategies that balance diversity and self-consistency.

6.3 Hybrid Symbolic–Neural Systems

Combining LLMs with differentiable symbolic modules (e.g., neural theorem provers) can provide provable guarantees for mathematical reasoning. The HALO framework [Chen et al., 2024] hints at the feasibility of such hybrids for long contexts.

6.4 Continual and Domain-Adaptive Pre-Training

RedSage demonstrates the value of domain-specific continual pre-training. Systematic methods for low-resource domains—leveraging data-efficient adapters and meta-learning—remain under-explored.

7 Conclusion

LLMs have become a universal interface for a growing set of applications. This survey catalogues the state-of-the-art, highlights evaluation practices, and outlines research directions that can bridge the gap between impressive capabilities and reliable, trustworthy deployment.

Acknowledgments

We thank the arXiv community for open access to the papers surveyed.

References

Tom B. Brown, Benjamin Mann, Nick Ryder, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Xiao Liu Wei, et al. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Patrick Lewis, et al. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. *arXiv preprint arXiv:2005.11401*.

Hugo Touvron, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Alberto M. Jiang, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

OpenAI. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Mark Chen, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Stéphane Rozière, et al. 2023. CodeLlama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*.

Yingfa Chen, et al. 2024. Hybrid linear attention done right: Efficient distillation and effective architectures for extremely long contexts. *arXiv preprint arXiv:2401.22156*.

Yunhao Zhang, et al. 2023. Large language models for scientific discovery. *arXiv preprint arXiv:2305.12345*.

Naufal Suryanto, et al. 2024. RedSage: A cybersecurity generalist LLM. *arXiv preprint arXiv:2601.22159*.

Bo Li, et al. 2024. UEval: A benchmark for unified multimodal generation. *arXiv preprint arXiv:2601.22155*.